Screencast: 24-dns.webm or 24-dns.mp4

LaUSAH REFERENCE - Chapter 16, DNS: The Domain Name System

RHEL7 Networking Guide, Chapter 15, DNS Servers (PDF in weekly materials)

Where does BIND rank? (from LAUSAH Fourth Edition)

Table 17.1 Some popular implementations of DNS

Name	Author	Source	Share[a]	Comments
BIND	ISC	isc.org	80.3%	Auth or caching
Microsoft DNS	Microsoft	microsoft.com	15.4%	Myriad sins
djbdns[b]	Dan Bernstein	tinydns.org	2.6%	Violates some RFCs
PowerDNS	PowerDNS BV	powerdns.com	0.7%	Auth only
NSD[c]	NLnet Labs	nlnetlabs.nl	< 0.1%	Auth only, very fast
Unbound	NLnet Labs	unbound.net	–	Caching only, fast

a. Market share from isc.org's July 2009 Internet Domain Survey
b. Also known as tinydns, which is the server component of the djbdns package
c. Originally designed for root and top-level domain servers; now in general use

Introduction to DNS

DNS associates hostnames with their respective IP addresses, so that when users want to connect to other machines on the network, they can refer to them by name, without having to remember IP addresses.

Use of DNS and FQDNs also has advantages for system administrators, allowing the flexibility to change the IP address for a host without affecting name-based queries to the machine. Conversely, administrators can shuffle which machines handle a name-based query.

DNS is normally implemented using centralized servers that are authoritative for some domains and refer to other DNS servers for other domains.

When a client host requests information from a nameserver, it usually connects to port 53. The nameserver then attempts to resolve the FQDN based on its resolver library, which may contain authoritative information about the host requested or cached data from an earlier query. If the nameserver does not already have the answer in its resolver library, it queries other nameservers, called root nameservers, to determine which nameservers are authoritative for the FQDN in question. Then, with that information, it queries the authoritative nameservers to determine the IP address of the requested host. If a reverse lookup is performed, the same procedure is used, except that the query is made with an unknown IP address rather than a name.

Nameserver Zones

On the Internet, the FQDN of a host can be broken down into different sections. These sections are organized into a hierarchy (much like a tree), with a main trunk, primary branches, secondary branches, and so forth. Consider the following FQDN:

bob.sales.example.com

When looking at how an FQDN is resolved to find the IP address that relates to a particular system, read the name from right to left, with each level of the hierarchy divided by periods (.). In this example, com defines the top level domain for this FQDN. The name example is a sub-domain under com, while sales is a sub-domain under example. The name furthest to the left, bob, identifies a specific machine hostname.

Except for the hostname, each section is called a zone, which defines a specific namespace. A namespace controls the naming of the sub-domains to its left. While this example only contains two sub-domains, an FQDN must contain at least one sub-domain but may include many more, depending upon how the namespace is organized.

Zones are defined on authoritative nameservers through the use of zone files (which describe the namespace of that zone), the mail servers to be used for a particular domain or sub-domain, and more. Zone files are stored on primary nameservers (also called master nameservers), which are truly authoritative and where changes are made to the files, and secondary nameservers (also called slave nameservers), which receive their zone files from the primary nameservers. Any nameserver can be a primary and secondary nameserver for different zones at the same time, and they may also be considered authoritative for multiple zones. It all depends on how the nameserver is configured.

Nameserver Types

There are two primary nameserver configuration types:

Authoritative

Authoritative nameservers answer to resource records that are part of their zones only. This category includes both primary (master) and secondary (slave) nameservers.

Recursive

Recursive nameservers offer resolution services, but they are not authoritative for any zone. Answers for all resolutions are cached in a memory for a fixed period of time, which is specified bythe retrieved resource record.

Most Frequently Used Types of Records
DNS record types (from LAUSAH Fourth Edition)

Table 17.6 DNS record types

	Type	Name	Function
Zone	SOA	Start Of Authority	Defines a DNS zone
	NS	Name Server	Identifies servers, delegates subdomains
Basic	A	IPv4 Address	Name-to-address translation
	AAAA	IPv6 Address	Name-to-IPv6-address translation
	PTR	Pointer	Address-to-name translation
	MX	Mail Exchanger	Controls email routing
Security and DNSSEC	DS	Delegation Signer	Hash of signed child zone's key-signing key
	DNSKEY	Public Key	Public key for a DNS name
	NSEC	Next Secure	Used with DNSSEC for negative answers
	NSEC3[a]	Next Secure v3	Used with DNSSEC for negative answers
	RRSIG	Signature	Signed, authenticated resource record set
	DLV	Lookaside	Nonroot trust anchor for DNSSEC
	SSHFP	SSH Fingerprint	SSH host key, allows verification via DNS
	SPF	Sender Policy	Identifies mail servers, inhibits forging
	DKIM	Domain Keys	Verify email sender and message integrity
Optional	CNAME	Canonical Name	Nicknames or aliases for a host
	SRV	Services	Gives locations of well-known services
	TXT	Text	Comments or untyped information[b]

a. The original NSEC system allows hackers handy with the **dig** command to easily list all of a zone's records. NSEC3 has fixed this weakness but is more expensive to compute; both are currently in use.

b. TXT records are increasingly being used to try out new ideas without having to get full IETF blessing for new record types. For example, SPF and DKIM records were first implemented as TXT records.

A - Address Records
CNAME - Canonical Name maps one name to another
MX - Mail eXchange record
NS - NameServer record
PTR - PoinTeR record, primarily for reverse resolution
SOA - Start Of Authority resource record
Others - SRV, TXT, AAAA

bind package information

Package name:
bind - With authoritative zones

Service control script:
/etc/rc.d/init.d/named (sysvinit)
/usr/lib/systemd/system/named.service (systemd)

Binary:
/usr/sbin/named

Config files:
[root@ct-dowdle /]# rpm -qc bind
/etc/logrotate.d/named
/etc/named.conf
/etc/named.iscdlv.key

```
/etc/named.rfc1912.zones
/etc/named.root.key
/etc/rndc.conf
/etc/rndc.key
/etc/sysconfig/named
/var/named/named.ca
/var/named/named.empty
/var/named/named.localhost
/var/named/named.loopback
```

Simplified view of how it works

/etc/named.conf states which zones a name server is authoritative for.
Example:

```
options {
        directory "/var/named";
};

zone "example.com." IN {
        type master;
        file "example.com.zone";
};
```

See "Example Zone Record" in this week's content to see a sample zone record.

Zone Transfers

A zone transfer is the process by which a DNS master communicates with one or more DNS slave servers to propagate new and updated zone information. With BIND you have to create an encryption key setup where your servers trust each other and accept transfers.

Common Mistakes to Avoid

It is very common for beginners to make mistakes when editing BIND configuration files. Be sure to avoid the following issues:

- Take care to increment the serial number when editing a zone file.
- If the serial number is not incremented, the master nameserver has the correct, new information, but the slave nameservers are never notified of the change and do not attempt to refresh their data of that zone.
- Be careful to use ellipses and semi-colons correctly in the /etc/named.conf file. An omitted semi-colon or unclosed ellipse section can cause named to refuse to start.
- Remember to place periods (.) in zone files after all FQDNs and omit them on hostnames.
- A period at the end of a domain name denotes a fully qualified domain name. If the period is omitted, then named appends the name of the zone or the $ORIGIN value to complete it.
- If a firewall is blocking connections from the named program to other nameservers, update your firewall.
- By default, BIND version 9 uses random ports above 1024 to query other nameservers. Some firewalls, however, expect all nameservers to communicate using only port 53. To force named to use port 53, add the following line to the options statement of /etc/named.conf:

- query-source address * port 53;

Hands On - Installing a caching-only nameserver in your student VM:

Inside your student VM as root do the following:
yum install bind (may already be installed)

Is bind set to run automatically? If not:

systemctl status named.service
systemctl enable --now named.service

Now to start using our own nameserver for DNS resolution:

nano -w /etc/resolv.conf
Comment out current line, add new line:
nameserver 127.0.0.1

Test it. Does it work?
How do you test?

When done with the homework:

There is a /public/homework7.txt on the course server. Just copy that to your ~/HOMEWORK/ directory on your student VM.